

Um Controlador Ótimo para o Gerenciamento Ativo de Filas

Michele M. de A. E. Lima Nelson L. S. da Fonseca José C. Geromel

Resumo—Este artigo introduz uma nova política de AQM que utiliza um controlador ótimo, H_2 . A síntese do controlador usa uma abordagem não racional, no qual os objetivos de estabilidade e desempenho são completamente expressos como desigualdades lineares de matrizes ou LMIs (*Linear Matrix Inequalities*). Nos resultados numéricos é demonstrado que o controlador estabiliza o sistema sob diferentes condições de rede.

Palavras-Chave—Controle de congestionamento, AQM, controladores ótimos, desigualdades lineares de matrizes (LMIs).

Abstract—This paper introduces a novel AQM policy that uses an H_2 optimal controller. The synthesis of the controller uses a non-rational approach, in which the stability and performance objectives of the system are completely expressed as Linear Matrix Inequalities (LMIs). The controller stabilizes the system under diverse network conditions.

Keywords—Congestion control, AQM, optimal controllers, Linear Matrix Inequalities (LMI).

I. INTRODUÇÃO

O TCP (*Transmission Control Protocol*) modifica a sua taxa de transmissão de acordo com a quantidade estimada de banda passante disponível. Tal modificação é governada pelo recebimento de reconhecimentos (ACKs - *acknowledgements*) enviados pelo receptor. Se três ACKs são recebidos para o mesmo pacote, ele é considerado perdido e a janela de transmissão é reduzida pela metade. Caso um reconhecimento não seja recebido depois de um certo intervalo de tempo (*timeout*), o pacote também é considerado perdido, e neste caso, a janela de transmissão é reduzida drasticamente para um pacote. Durante o período decorrido entre a perda do pacote e a sua detecção, inúmeros outros pacotes transmitidos também podem ter sido perdidos, aumentando ainda mais o congestionamento e o desperdício de banda passante. Para amenizar este problema, foram propostos os mecanismos de Gerenciamento Ativo de Filas ou AQM (*Active Queue Management*).

A idéia por trás de AQM é a notificação antecipada do congestionamento incipiente, de forma a permitir que os TCP emissores possam reduzir sua taxa de transmissão antes que as filas fiquem cheias, evitando assim, a degradação do desempenho do TCP.

RED - *Random Early Detection* [1], é o algoritmo de AQM recomendado pelo IETF para a Internet [2]. RED foi proposto para prevenir o congestionamento, garantir um limite para o

tamanho médio das filas, evitar o fenômeno de sincronização global e prevenir a penalização dos tráfegos em rajada.

RED estima o tamanho médio da fila, que é comparado com dois *thresholds* min_{th} e max_{th} . Se o valor estiver abaixo de min_{th} o algoritmo está na zona normal de operação e nenhum pacote é descartado/marcado. Caso o valor esteja entre os dois *thresholds*, cada pacote que chega é descartado/marcado com uma probabilidade p_a , que cresce linearmente com o tamanho médio da fila estimado. Se o valor estiver acima de max_{th} , todos os pacotes que chegam são descartados. Determinar os parâmetros de RED é um desafio. Quando os valores dos *thresholds* não são corretamente definidos, RED pode vir a se comportar até pior que a política *tail drop*. Quando o valor de min_{th} é muito baixo, o enlace pode ser subutilizado e quando o valor de max_{th} é muito alto, aumenta-se desnecessariamente o atraso.

O desempenho de RED com TCP é suscetível ao número de conexões ativas. Quando existe um grande número de conexões ativas, o tráfego agregado é em rajadas, causando grandes oscilações no tamanho da fila, e conseqüentemente, aumentando a variação do retardo (*jitter*) e a subutilização do enlace. Além disto, quando o valor da probabilidade de descarte/marcação varia muito em um curto período, RED falha em marcar os pacotes de forma aleatória, causando o fenômeno de sincronização global, que é a redução das janelas de transmissão de várias conexões ao mesmo tempo em resposta ao descarte/marcação de pacotes. Outrossim, quando o tamanho médio da fila é maior que max_{th} , todos os pacotes são descartados causando um decréscimo significativo na vazão.

Com o intuito de superar as dificuldades de determinar corretamente os parâmetros de RED, vários estudos baseados em heurísticas e simulações vêm sendo desenvolvidos. Contudo, tais estudos não garantem que um ponto de equilíbrio seja atingido, nem tampouco garantem a estabilidade do tamanho da fila [3], [4]. Por outro lado, investigações vêm sendo conduzidas para derivar configurações para RED de uma forma mais sistemática [5], [6].

Para se projetar e desenvolver políticas de AQM que garantam estabilidade em torno de um ponto de equilíbrio, algumas abordagens baseadas em Otimização e Teoria de Controle vem sendo propostas. No primeiro caso, busca-se caracterizar as condições de equilíbrio que podem ser obtidas de acordo com as condições da rede. Nas políticas baseadas em Teoria de Controle, o controle de congestionamento é visto como um sistema de controle de retroalimentação. O objetivo é compreender a dinâmica do congestionamento, para se poder garantir estabilidade no tamanho da fila. Em [7], foi modelado

Colegiado de Informática - (UNIOESTE). Doutoranda no Instituto de Computação - (UNICAMP)

Instituto de Computação - Universidade Estadual de Campinas - (UNICAMP)

Faculdade de Engenharia Elétrica e de Computação - Universidade Estadual de Campinas - (UNICAMP)

o comportamento de uma fila AQM, e em [8], os autores propuseram uma política de AQM que usa um controlador do tipo PI (*Proportional Integrator*) e compararam o seu desempenho com RED.

Este artigo apresenta uma nova política de AQM, H2-AQM, que utiliza um controlador ótimo. Uma simplificação do modelo dinâmico não-linear do comportamento TCP apresentado em [9] é utilizado para projetar o controlador. O modelo é linearizado e o sistema obtido é linear com atraso constante no tempo. O sistema resultante é analisado e é determinada a saída desejada que dê o desempenho esperado e, posteriormente, um controlador que produza a solução ótima para atingir o desempenho alvo é desenvolvido.

A contribuição principal deste trabalho é a nova abordagem de se utilizar controladores não racionais para o sistema, superando-se a principal dificuldade de se projetar controladores racionais para sistemas com atraso, que é a incorporação da matriz de multiplicação utilizada para provar a estabilidade com relação à parte com atraso do sistema no projeto do problema [10]. Além disto, a estabilidade e os objetivos de desempenho do sistema de AQM são completamente expressos e solucionados através de desigualdades matriciais lineares ou LMIs (*Linear Matrix Inequalities*).

Este artigo é organizado da seguinte forma: a seção II descreve os trabalhos correlatos. A seção III mostra o modelo do TCP utilizado na definição do sistema de AQM proposto. Na seção IV, é desenvolvida a política de gerenciamento ativo de filas, H2-AQM, que utiliza um controlador não-racional para o sistema. Na seção V, os resultados numéricos são apresentados. Finalmente, na seção VI, as conclusões são delineadas.

II. TRABALHOS RELACIONADOS

Nas políticas de AQM baseadas na Teoria de Otimização almeja-se maximizar as taxas de transmissão das fontes, de forma a que haja um compartilhamento equalitário entre as mesmas da banda passante disponível [11], [12]. Funções de utilidade que descrevam o fluxo agregado são definidas para identificar as condições de equilíbrio para um dado estado da rede.

Na versão dual deste problema, a taxa de perda é tratada como a variável dual. O objetivo é obter uma taxa de perda mínima que garanta as taxas de transmissão máximas. A política REM - *Random Exponential Marking* é apresentada como a solução para este problema [13]. REM expressa as medidas de congestionamento através de uma quantidade denominada, preço, computada para cada enlace utilizando informações locais, e informada aos nós fontes através do descarte/marcação dos pacotes.

As políticas baseadas na Teoria de Controle consideram a natureza intrínseca de retroalimentação do sistema de controle de congestionamento, sendo a taxa de transmissão dos nós fontes ajustada de acordo com o estado de congestionamento, que é determinado pela ocupação da fila. A notificação de congestionamento para os nós fontes é feito através do descarte/marcação dos pacotes. Desta forma, os controladores são responsáveis por determinar qual a probabilidade de

descarte/marcação adequada que estabilize o tamanho da fila independentemente das variações das condições da rede, o que é de grande importância para evitar *jitter* e baixa utilização do enlace.

SRED - *Stabilized RED* objetiva estabilizar o tamanho da fila em torno de um valor de referência independentemente do número de fluxos ativos. Para tanto, SRED estima o número de conexões TCP ativas e determina qual deve ser a probabilidade de descarte/marcação para este número de fluxos [14]. DRED - *Dynamic RED* possui o mesmo objetivo de SRED, sem, no entanto, estimar a quantidade de fluxos ativos. DRED utiliza um controlador proporcional, onde o ganho é determinado de forma empírica e baseado em simulações [15].

PI - *Proportional Integrator* [16], utiliza um controlador de mesmo nome e baseia-se em um modelo simplificado da dinâmica do comportamento do TCP apresentado em [7]. Como a planta apresentada nesta proposta não representa toda a dinâmica do sistema, faz-se necessário identificar qual a sensibilidade deste sistema, bem como estabelecer as condições da rede para as quais o controlador consegue estabilizá-lo.

O controlador H2-AQM, apresentado neste trabalho, utiliza mesmo modelo da dinâmica do comportamento do TCP que foi usado para derivar o controlador PI-AQM. No entanto, a planta utilizada no desenvolvimento do controlador H2-AQM representa a dinâmica do sistema em detalhes, garantindo a sua estabilidade independente das condições de rede. Uma outra diferença é que H2-AQM usa técnicas de controle ótimo ao invés de controladores clássicos como PI. Além disto, a estabilidade e os objetivos de desempenho do sistema de AQM são completamente expressos e solucionados como desigualdades matriciais lineares, o que significa que os parâmetros do controlador podem ser calculados através da solução de um simples problema convexo.

III. MODELO DINÂMICO PARA O COMPORTAMENTO DO TCP

Um modelo dinâmico para o comportamento do TCP baseado em análises de fluido e equações diferenciais estocásticas foi introduzido em [7]. Uma versão simplificada deste modelo, que ignora o mecanismo de *timeout* é dada por [8]:

$$\dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)}{2} \frac{W(t - R(t))}{R(t - R(t))} p(t - R(t)); \quad (1)$$

$$\dot{q}(t) = -C(t) + \frac{N(t)}{R(t)} W(t) + \omega_q(t); \quad (2)$$

$$R(t) = \frac{q(t)}{C(t)} + T_p; \quad (3)$$

onde:

$W(t)$: é a média do tamanho da janela TCP em pacotes;

$q(t)$: é o tamanho da fila em pacotes;

$\omega_q(t)$: é o ruído gerado pelo tráfego UDP;

$R(t)$: é o tempo total de viagem (RTT) em segundos;

$C(t)$: é a capacidade do enlace em pacotes/segundo;

T_p : é o tempo de propagação em segundos;

$N(t)$: é o fator de carga em número de conexões TCP;

$p(t)$: é a probabilidade de marcação/descarte de pacotes;

A equação (1) descreve a dinâmica da janela TCP, onde o primeiro termo corresponde ao crescimento aditivo da janela e o segundo termo modela o decréscimo multiplicativo da janela. A equação (2), modela o tamanho da fila como a diferença entre a taxa de chegada, $NW/R + \omega_q(t)$ e a capacidade do enlace, C .

A equação (2) difere da equação apresentada em [9] pela presença do termo $\omega_q(t)$, que foi incluído para que a variação no tamanho da fila também leve em consideração os pacotes referentes aos fluxos UDP. Estes fluxos são mal comportados ou não adaptativos, o que significa que não diminuem sua taxa de transmissão ao serem notificados da existência de congestionamento.

As equações (1) e (2), são linearizadas em torno do seu ponto de equilíbrio, com $\omega_q(t) = 0$, para que se possa analisar e projetar um controlador AQM. Para tanto, assume-se que o número de conexões TCP e a capacidade do enlace são constantes, ou seja, $N(t) \equiv N$ e $C(t) \equiv C$. Seja, então, (W, q) o estado e p a entrada, o ponto de equilíbrio (W_0, q_0, p_0) é obtido através da solução de $\dot{W}(t) = 0$ e $\dot{q}(t) = 0$, resultando em:

$$W_0 = \sqrt{\frac{2}{p_0}} = \frac{R_0 C}{N} = \frac{q_0 + CT_p}{N}; \quad (4)$$

$$q_0 = N \sqrt{\frac{2}{p_0}} - CT_p = CR_0 - CT_p; \quad (5)$$

$$p_0 = \frac{2N^2}{(R_0 C)^2} = \frac{2N^2}{(q_0 + CT_p)^2}; \quad (6)$$

Linearizando-se (1) e (2) em torno do ponto de equilíbrio (W_0, q_0, p_0) se obtém:

$$\begin{aligned} \dot{x}_1(t) &= -\frac{N}{R_0^2 C} (x_1(t) + x_1(t - R_0)) - \frac{R_0 C^2}{2N^2} u(t - R_0) \\ &\quad - \frac{1}{R_0^2 C} (x_2(t) - x_2(t - R_0)); \end{aligned} \quad (7)$$

$$\dot{x}_2(t) = \frac{N}{R_0} x_1(t) - \frac{1}{R_0} x_2(t);$$

onde:

$$\begin{aligned} x_1(t) &\doteq W(t) - W_0; \\ x_2(t) &\doteq q(t) - q_0; \\ u(t) &\doteq p(t) - p_0; \end{aligned}$$

IV. PROJETO DE UM CONTROLADOR ÓTIMO PARA AQM

Nesta seção, o sistema de controle de congestionamento (7) é representado na forma de estado de espaço como um sistema linear com atraso e uma abordagem não racional é usada para derivar o controlador ótimo H_2 . A síntese do controlador é baseada nos resultados apresentados em [10], onde os projetos de controladores para sistemas lineares com atraso são expressos e solucionados como desigualdades matriciais lineares, LMIs (*Linear Matrix Inequalities*).

A dinâmica TCP descrita em (7) pode ser analisada em termos de parâmetros de rede tais como o número de fluxos TCP, N , o tempo total de viagem (RTT), R_0 , a capacidade

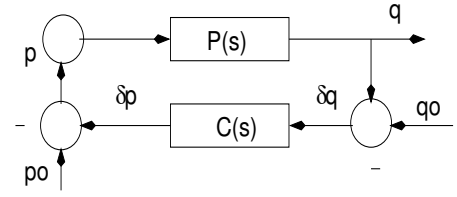


Fig. 1. Controle de Congestionamento como um Sistema de Controle com Retroalimentação

do enlace, C , e em termos da natureza de retroalimentação do AQM [8]. A função do controlador AQM, $C(s)$, é marcar os pacotes com probabilidade p , como função do tamanho da fila medido q , como também estabilizar a planta, denotado pela função de transferência $P(s)$, irracional em s , que descreve como a probabilidade de marcação/descarte de pacotes afeta o tamanho da fila. A Figura 1 apresenta o sistema de congestionamento como um sistema de controle com retroalimentação.

A planta $P(s)$ representa a dinâmica do sistema com maior detalhamento que a planta apresentada em [8], onde a planta é simplificada de forma a isolar as contribuições devido ao atraso no resíduo $\Delta(s)$. Este resíduo é tratado como uma dinâmica não modelada do sistema.

A vantagem de se representar completamente a dinâmica do sistema na planta $P(s)$ está no fato de que a estabilização da planta pelo controlador $C(s)$, implica na estabilidade do sistema.

Na abordagem adotada por [8], para que o sistema seja estável o controlador $C(s)$ também tem que estabilizar o resíduo $\Delta(s)$ e estabelecer um limite superior para $\Delta(s)V(s)$, onde $V(s)$ é a função de sensibilidade do sistema. Isto significa que devem ser identificados valores de N , R_0 e C , para os quais o controlador consegue estabilizar o sistema.

O sistema linear apresentado em (7) pode ser expresso no estado de espaço pelas seguintes equações que descrevem um sistema linear com atraso contínuo no tempo:

$$\begin{aligned} \dot{x}(t) &= A_0 x(t) + A_1 x(t - R_0) + B_w w(t) \\ &\quad + B_u u(t); \end{aligned} \quad (8)$$

$$\begin{aligned} z(t) &= C_z x(t) + D_{zu} u(t); \\ y(t) &= C_y x(t - R_0) + D_{yw} w(t); \end{aligned}$$

onde $x(t)$ é o vetor de estado; $u(t)$ é a entrada a ser controlada e representa a probabilidade $p(t)$; $w(t)$ é o ruído externo produzido pelos fluxos UDP; $z(t)$ é a saída de referência, ou seja, saída esperada para o sistema e $y(t)$ a saída obtida para o sistema. O atraso presente em (7) é apresentado em (8) na saída $y(t)$.

Considere que o sistema descrito em (8) esteja conectado com o controlador:

$$\begin{aligned} \dot{\hat{x}}(t) &= \hat{A}_0 \hat{x}(t) + \hat{A}_1 \hat{x}(t - R_0) + \hat{B} y(t); \\ u(t) &= \hat{C}_0 \hat{x}(t) + \hat{C}_1 \hat{x}(t - R_0) + \hat{D} y(t); \end{aligned} \quad (9)$$

Este controlador também pode ser expresso no domínio da frequência pela função de transferência não racional:

$$C(s) = (\hat{C}_0 + \hat{C}_1 e^{-sR_0})(sI - \hat{A}_0 - \hat{A}_1 e^{-sr_0})^{-1} \hat{B} + \hat{D}; \quad (10)$$

O controlador (9) foi escolhido de forma a reproduzir a estrutura da planta do sistema (8). O objetivo é determinar as matrizes do controlador (9) que estabilizem (8) e minimizem certas medidas na saída de referência $z(t)$. É necessário, então, definir quais são os objetivos de desempenho desejados para a saída $z(t)$ e o que deve ser medido na saída obtida do sistema, $y(t)$. Isto implica que os objetivos de desempenho para a política de AQM projetada devem estar representados no controlador $C(s)$.

Os objetivos de desempenho para uma política de AQM como sugeridos em [8] são: prevenção e controle do congestionamento, eficiente utilização da fila e a garantia de baixo retardo e de baixa variação no retardo. Utilizar a fila de forma eficiente significa evitar que existam períodos em que ela esteja cheia ou vazia. No primeiro caso resulta na perda de pacotes e retransmissões desnecessárias, como também penaliza tráfegos em rajada, enquanto o segundo caso leva à subutilização do enlace. Para que se tenham baixos valores de retardo deve-se ter tamanhos pequenos de fila, o que pode gerar a subutilização do enlace. Além disto, é desejável que grandes variações no tamanho da fila sejam evitados para prevenir variações no retardo (*jitter*), o que é prejudicial para algumas aplicações de tempo real. Assim, o valor ideal da probabilidade de descarte/marcação deve tentar ao mesmo tempo maximizar as taxas de transmissão e minimizar o tamanho da fila sujeito às condições da rede, de forma a evitar perdas de pacotes desnecessárias. Para atingir tais objetivos, as matrizes do sistema (8), são definidas como:

$$A_0 = \begin{bmatrix} -\frac{N}{R_0^2 C} & -\frac{1}{R_0^2 C} \\ \frac{N}{R_0} & -\frac{1}{R_0} \end{bmatrix}, \quad B_w = \begin{bmatrix} 0 & 0 \\ 0.2C & 0 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} -\frac{N}{R_0^2 C} & \frac{1}{R_0^2 C} \\ 0 & 0 \end{bmatrix}, \quad B_u = \begin{bmatrix} -\frac{R_0 C^2}{2N^2} \\ 0 \end{bmatrix},$$

$$C_z = \begin{bmatrix} 0 & 1 \\ \frac{N}{R_0} & -\frac{1}{R_0} \\ 0 & 0 \end{bmatrix}, \quad D_{zu} = \begin{bmatrix} 0 \\ 0 \\ 0.5 \end{bmatrix};$$

$$C_y = [0 \ 1], \quad D_{yw} = [0 \ 0.02C]$$

As matrizes A_0 , A_1 e B_u são obtidas diretamente da linearização do sistema. A_0 representa os termos do sistema sem atraso, enquanto A_1 representa os termos com atraso. A primeira coluna destas matrizes corresponde a $\frac{\partial f_1}{\partial x_i}$, e a segunda corresponde a $\frac{\partial f_2}{\partial x_i}$, com $i = 1, 2$. B_u contém $\frac{\partial f_1}{\partial u}$ na primeira coluna, e $\frac{\partial f_2}{\partial u}$ na segunda coluna, onde f_1 é o lado direito da Equação (1) e f_2 o lado direito da Equação (2), $x_1 = W$, $x_2 = q$ e $u = p$.

B_w controla a quantidade de ruído no sistema, que é gerado pelos fluxos UDP. O valor considerado permite que os fluxos UDP utilizem até 20% da capacidade do enlace. Este valor é uma margem de tolerância satisfatória, dado que 95% dos bytes transmitidos na Internet são gerados pelo TCP [17].

C_z representa o objetivo do projeto, que é tentar obter o equilíbrio entre maximizar as taxas de transmissão e ao mesmo tempo minimizar o tamanho da fila, como também minimizar a sua variação. A primeira linha está relacionada com o tamanho da fila e a segunda com a sua variação. D_{zu} , pondera o valor da probabilidade de descarte/marcação na saída. Diferentes valores foram verificados para esta ponderação, variando de 0,3 a 0,9. Os resultados obtidos foram bastante similares, sendo 0,5 o valor adotado.

C_y indica que o valor de interesse medido na saída é o tamanho da fila no RTT anterior. Finalmente, D_{yw} , pondera o ruído que é medido na saída, que é geralmente 10% do valor presente na matriz B_w .

Seja $\bar{x}(t)$ o vetor de estado aumentado que contém o vetor de estado $x(t)$ e o vetor de estado do controlador $\hat{x}(t)$:

$$\bar{x}(t) = \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix}; \quad (11)$$

A conexão do sistema (8) com o controlador (9) resulta no sistema linear com atraso:

$$\dot{\bar{x}}(t) = A_0 \bar{x}(t) + A_1 \bar{x}(t - R_0) + B_w w(t); \quad (12)$$

$$z(t) = C_0 \bar{x}(t) + C_1 \bar{x}(t - R_0) + D w(t);$$

onde:

$$A_0 = \begin{bmatrix} A_0 & B_u \hat{C}_0 \\ 0 & \hat{A}_0 \end{bmatrix};$$

$$A_1 = \begin{bmatrix} A_1 + B_u \hat{D} C_y & B_u \hat{C}_1 \\ \hat{B} C_y & \hat{A}_1 \end{bmatrix};$$

$$B = \begin{bmatrix} B_w + B_u \hat{D} D_{yw} \\ \hat{B} D_{yw} \end{bmatrix};$$

$$C_0 = [C_z \ D_{zu} \hat{C}_0];$$

$$C_1 = [D_{zu} \hat{D} C_y \ D_{zu} \hat{C}_1];$$

$$D = D_{zu} \hat{D} D_{yw};$$

Para garantir a estabilidade do sistema (12), o Teorema 4-b apresentado em [10] é usado. Este teorema especifica que um sistema como (12) é assintoticamente estável e $\|H_{wz}(s)\|_2^2 < \gamma$, se existem matrizes simétricas e definidas positivas W , Y_0 e X_j , e matrizes F , R , L_j e Q_j , com $j = 0, 1$, tais que os seguintes LMIs tenham uma solução factível:

$$\begin{bmatrix} A_0 + A_0^T + X_1 & (\bullet)^T & (\bullet)^T \\ A_1^T & -X_1 & (\bullet)^T \\ C_0 & C_1 & -I \end{bmatrix} < 0 \quad (13)$$

$$\begin{bmatrix} W & (\bullet)^T \\ B & P_0 \end{bmatrix} > 0, \quad \text{trace}(W) < \gamma \quad (14)$$

onde A_0 , A_1 , B , C_0 , C_1 , D e P_0 são dadas por:

$$A_0 = \begin{bmatrix} A_0 X_0 + B_u L_0 & A_0 \\ Q_0 & Y_0 A_0 \end{bmatrix},$$

$$\begin{aligned} \mathbf{A}_1 &= \begin{bmatrix} A_1 X_0 + B_u L_1 & A_1 \\ Q_1 & Y_0 A_1 + F C_y \end{bmatrix}, \\ \mathbf{B} &= \begin{bmatrix} B_w \\ Y_0 B_w + F D_{yw} \end{bmatrix}, \\ \mathbf{C}_0 &= [C_z X_0 + D_{zu} L_0 \quad C_z], \\ \mathbf{C}_1 &= [D_{zu} L_1 \quad 0], \\ \mathbf{P}_0 &= \begin{bmatrix} X_0 & I \\ I & Y_0 \end{bmatrix}; \end{aligned}$$

Este problema de programação convexa foi resolvido numericamente utilizando a rotina LMISol [18]. Os parâmetros de rede utilizados foram os mesmos fornecidos em [9], [8], ou seja, $N = 60$ sessões TCP, $R_0 = 0,246$ segundos e $C = 3750$ pacotes/segundos, o que corresponde a um enlace de 15 Mb/s e pacotes com tamanho médio de 500 bytes. Como foi obtida uma solução factível para os LMIs ((13) e (14), pode-se afirmar que o sistema (12) é estável.

Obtida uma solução factível, o próximo passo é determinar os parâmetros do controlador (9). Primeiramente, matrizes arbitrárias U_0 e V_0 devem ser escolhidas de forma a satisfazer $V_0 U_0 = I - Y_0 X_0$. As matrizes utilizadas foram $U_0 = X_0$ e $V_0 = X_0^{-1} - Y_0$. Assim, os parâmetros do controlador podem ser determinados por:

$$\begin{bmatrix} \hat{A}_0 & \hat{A}_1 & \hat{B} \\ \hat{C}_0 & \hat{C}_1 & \hat{D} \end{bmatrix} = \mathcal{K} \cdot \mathcal{M} \cdot \mathcal{N}. \quad (15)$$

onde:

$$\begin{aligned} \mathcal{K} &= \begin{bmatrix} V_0^{-1} & -V_0^{-1} Y_0 B_u \\ 0 & I \end{bmatrix}; \\ \mathcal{M} &= \begin{bmatrix} Q_0 - Y_0 A_0 X_0 & Q_1 - Y_0 A_1 X_0 & F \\ L_0 & L_1 & 0 \end{bmatrix}; \\ \mathcal{N} &= \begin{bmatrix} U_0^{-1} & 0 & 0 \\ 0 & U_0^{-1} & 0 \\ -C_{y0} X_0 U_0^{-1} & -C_{y1} X_0 U_0^{-1} & I \end{bmatrix}; \end{aligned}$$

Na solução ótima obtida, as matrizes \hat{A}_1 e \hat{C}_1 são aproximadamente zero, sendo portanto ignoradas. Como resultado tem-se o cancelamento do termo de atraso do sistema, e o controlador torna-se racional. O cancelamento do termo de atraso, quando possível, é a estratégia global ótima para solucionar o problema de minimização da norma H_2 [10]. O controlador $C_{H2}(s)$ representado por sua função de transferências no domínio da frequência é dado por:

$$C_{H2}(s) = \frac{0,82426s}{s + 8,977e004}; \quad (16)$$

Observe que $C_{H2}(s)$, a função de transferência apresentada em (16) é de ordem 1 e não de ordem 2 como esperado, dado que com a fatoração da função de transferência, o fator $(s + 3,865)$ presente tanto no numerador quanto no denominador, foi cancelado.

H2-AQM-ProbabilityFunction()

```

N ← NumberActiveFlows();
p0 ← 2 * N^2 / (R0 * C)^2;
q0 ← C * (R0 - Tp);
p ← a * (q - 2 * q0 + qold) - b * pold + p0 * (1 + b);
pold ← p;
qold ← q;

```

Fig. 2. Algoritmo para o cálculo da probabilidade em H2-AQM

Para que C_{H2} possa ser implementado, é necessário que seja escolhida uma frequência de amostragem f_s , para que uma representação no domínio- z possa ser obtida. A frequência escolhida foi $f_s = 2625 Hz$, ou 70% do valor da capacidade do enlace, C , utilizada para a obtenção do ponto de equilíbrio. O controlador C_{H2} no domínio- z é dado por:

$$C_{H2}(z) = \frac{a(z+1)}{z+b} = \frac{8,6742e-006(z+1)}{z+0,8895}; \quad (17)$$

A função de transferência entre $\delta p = p - p_0$ e $\delta q = q - q_0$, apresentada em (17), pode ser convertida em uma equação de diferenças no tempo discreto kT , onde $T = \frac{1}{f_s}$:

$$\delta p(kT) = a[\delta q(kT) + \delta q((k-1)T)] - b\delta p((k-1)T); \quad (18)$$

O algoritmo para o cálculo da probabilidade de descarte/marcação em H2-AQM é bastante simples, e é executado a cada intervalo de amostragem $1/f_s$ (Figura 2). Inicialmente, o algoritmo estima a quantidade de fluxos TCP ativos. Depois, são calculados os valores de referência p_0 e q_0 , em torno dos quais o sistema deve estabilizar. Finalmente, o valor da probabilidade é calculado baseado em (18). O algoritmo necessita de duas variáveis auxiliares: q_{old} e p_{old} , que armazenam os valores de q e p respectivamente no intervalo anterior.

V. RESULTADOS NUMÉRICOS

Em [8], a superioridade da política PI-AQM em relação a RED foi evidenciada. Por este motivo, nesta seção H2-AQM é comparado apenas com PI-AQM. O software Simulink foi utilizado para modelar e simular ambos os controladores. O controle de congestionamento foi modelado como um sistema de retroalimentação onde pode-se escolher o controlador que determina a probabilidade de descarte/marcação $p(t)$. Diferentes valores de N, C, T_p e R_0 foram utilizados nas simulações. O tráfego foi composto tanto por fluxos TCP quanto UDP. Os parâmetros de rede utilizados para se obter o ponto de equilíbrio foram: $N = 60$ fluxos TCP, $T_p = 0,2$ segundos, $R_0 = 0,246$ segundos e $C = 3750$ pacotes/segundos. O tráfego UDP, $\omega_q(t)$, foi representado por um ruído branco, e pode ser igual a até 20% da capacidade do enlace. A condição inicial para o sistema foi definido como $W(0) = 1$ e $q(0) = 0$.

Primeiramente verificou-se o impacto do aumento do número de conexões ativas no sistema. Os valores de C, R_0 e

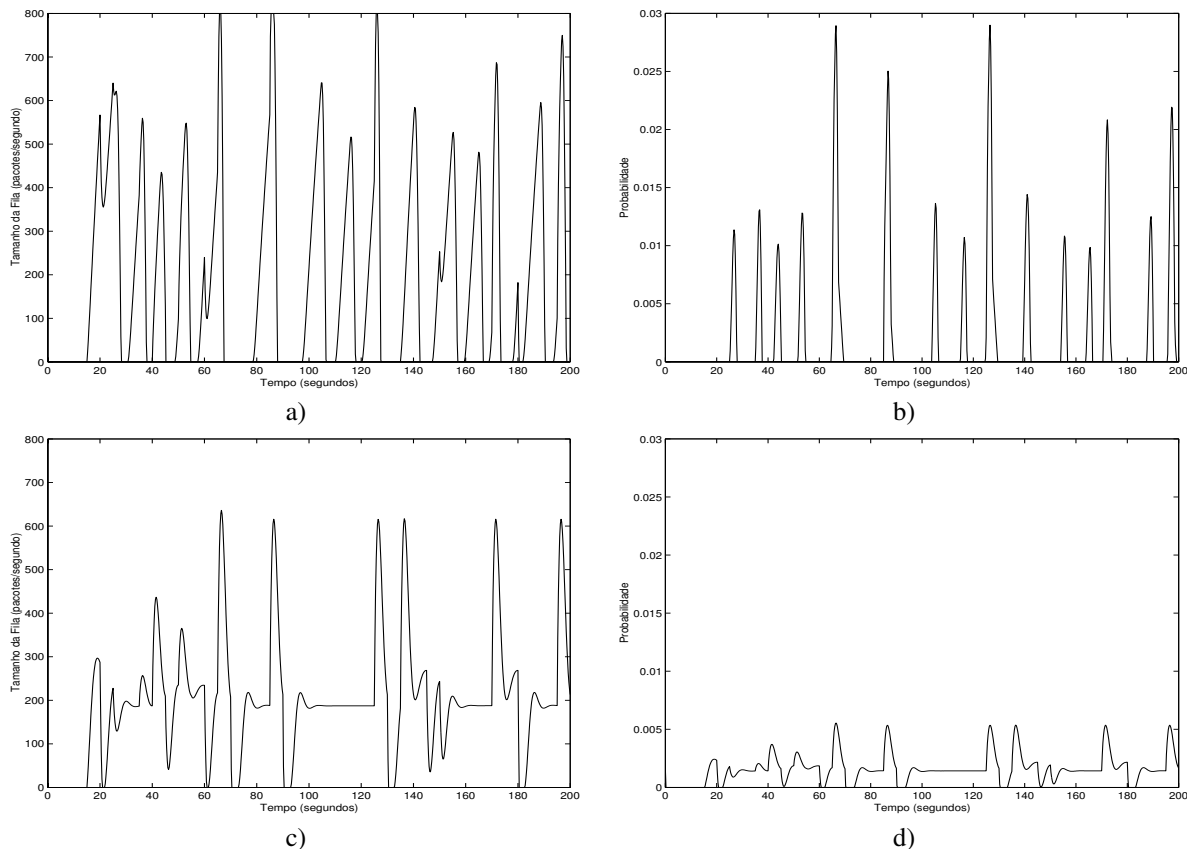


Fig. 3. Simulação com $N = 60$ fluxos TCP, $R_0 = 0.6$ segundos $T_p = 0.55$ segundos e $C = 3750$ pacotes/segundo. a) Fila PI-AQM b) Probabilidade PI-AQM, c) Fila H2-AQM, d) Probabilidade H2-AQM

T_p foram mantidos iguais aos valores iniciais e o número de fluxos TCP foi incrementado gradualmente até 600 fluxos. Foi verificado que ambos os controladores estabilizam o sistema, mesmo com um grande número de fluxos ativos.

O impacto do produto *Banda X Atraso* no sistema também foi investigado. Primeiramente, C e N foram mantidos iguais aos valores iniciais e o valor de T_p foi gradualmente incrementado e o valor de R_0 foi dado por $R_0 = T_p + 0,05$. Foi verificado que para valores de T_p maiores que 0,35 segundos, o controlador PI-AQM tem dificuldades de estabilizar o sistema. O tamanho da fila e o valor da probabilidade oscilam consideravelmente. Para valores de T_p maiores que 0,55 segundos, PI-AQM não estabiliza o sistema. As oscilações são muito intensas, causando grandes valores de *jitter*. H2-AQM, apesar de também apresentar pequenas oscilações nestas condições, ainda consegue estabilizar o sistema. (Figura 3).

Na segunda etapa, os valores de T_p e N foram mantidos constantes e iguais aos valores iniciais, e foram feitas várias simulações aumentando-se o valor de C . Quando o valor de C foi aproximadamente igual ao dobro do valor inicial, PI-AQM já exibe dificuldades em estabilizar o sistema, apresentando oscilações no tamanho da fila e na probabilidade de descarte/marcação. Quando o valor de C é igual a 8000 pacotes/segundo, PI-AQM definitivamente falha em estabilizar o sistema (Figura 4-a e 4-b). As oscilações apresentadas são muito intensas. Na Figura 4, pode-se verificar que H2-AQM estabiliza o sistema sob as mesmas condições. Os resultados da

simulação indicam que PI não consegue estabilizar o sistema quando se tem um número muito pequeno de conexões ativas comparado com o produto *Banda X Atraso*, enquanto H2-AQM, ao contrário, consegue.

Os resultados apresentados eram de certa forma esperados. PI-AQM utiliza uma planta simplificada do sistema, sendo limitadas as condições para as quais ele consegue estabilizar o sistema. Conseqüentemente, PI-AQM não garante a estabilidade do sistema para condições adversas. Por outro lado, H2-AQM, garante a estabilidade do sistema e também produz valores ideais para a probabilidade de descarte/marcação sob diferentes condições da rede, de forma a evitar perdas desnecessárias de pacotes.

VI. CONCLUSÕES

Este artigo introduziu um controlador ótimo para um sistema de AQM retroalimentado tanto por fluxos TCP quanto por fluxos UDP. H2-AQM considera um detalhada descrição do sistema, garantindo sua estabilização sob diferentes condições da rede. Apesar de ter sido usada uma abordagem não racional para sintetizar o controlador, o controlador obtido foi racional, dado que o cancelamento dos termos de atraso do sistema foi possível, obtendo a solução ótima para a minimização da norma H_2 do problema. Adicionalmente, o controlador obtido é de ordem um e não de ordem dois, como esperado, sendo assim, um controlador mais simples e conseqüentemente, permitindo que um algoritmo

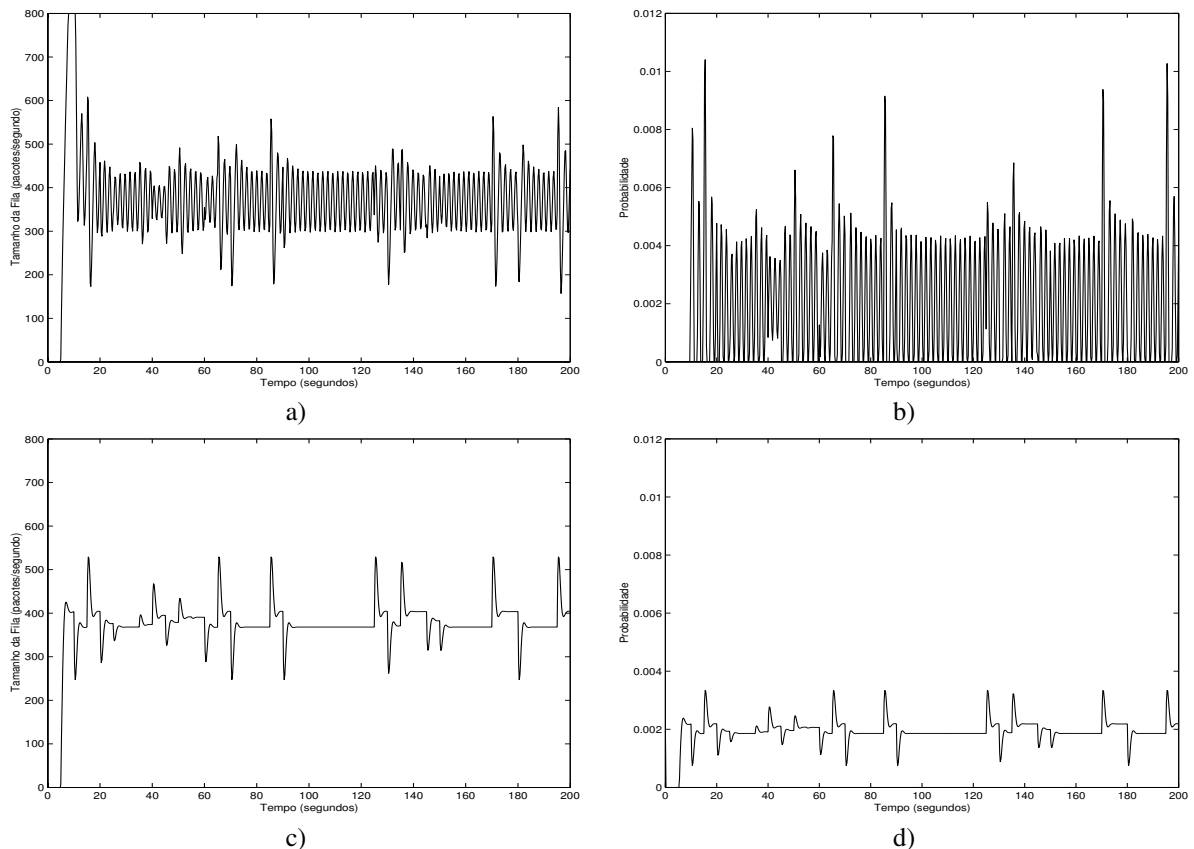


Fig. 4. Simulação com $N = 60$ fluxos TCP, $R_0 = 0,246$ segundos $T_p = 0,2$ segundos e $C = 8000$ pacotes/segundo. a) Fila PI-AQM b) Probabilidade PI-AQM, c) Fila H2-AQM, d) Probabilidade H2-AQM

muito simples seja desenvolvido. Foi demonstrado através das simulações que H2-AQM apresentou superou PI-AQM em atingir os objetivos de desempenho de AQM. H2-AQM está sendo implementado no simulador de redes NS, de forma a permitir a verificação do seu desempenho em um ambiente dinâmico de redes, como também permitir a comparação com outras políticas de AQM.

REFERÊNCIAS

- [1] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, 1993.
- [2] B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang, "Recommendations on queue management and congestion avoidance in the Internet," RFC 2309, apr 1998.
- [3] W. Feng, "Improving Internet congestion control and queue management algorithms," Ph.D. dissertation, University of Michigan, USA, 1999.
- [4] W. Feng, D. D. Kandlur, D. Saha, and K. G. Shin, "A self-configuring RED gateway," in *Proc. IEEE INFOCOM'99*, vol. 3, New York, NY, Mar. 1999, pp. 1320–1328.
- [5] V. Firoiu and M. Borden, "A study of active queue management for congestion control," in *Proc. IEEE INFOCOM'00*, 2000, pp. 1435–1444.
- [6] P. Kuusela, P. Lassila, and J. Virtamo, "Stability of TCP-RED congestion control," in *Proc. ITC-17*, Salvador da Bahia, Brasil, dec 2001, pp. 655–666.
- [7] V. Misra, W.-B. Gong, and D. F. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," in *Proc. ACM SIGCOMM'00*, 2000, pp. 151–160.
- [8] C. V. Hollot, V. Misra, D. Towsley, and W. Gong, "Analysis and design of controllers for AQM routers supporting TCP flows," 2002, to appear in special issue of IEEE Trans. Automat. Cont. on Systems and Control Methods for Communication Networks.
- [9] C. V. Hollot, V. Misra, D. F. Towsley, and W. Gong, "A control theoretic analysis of RED," in *Proc. IEEE INFOCOM'01*, 2001, pp. 1510–1519.
- [10] M. C. Oliveira and J. C. Geromel, "Synthesis of Non-Rational controllers for linear delay systems," 2002, to be published in Automatica.
- [11] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.
- [12] S. Kunniyur and R. Srikant, "End-to-end congestion control schemes: Utility functions, random losses and ECN marks," in *Proc. IEEE INFOCOM'00*, 2000, pp. 1323–1332.
- [13] S. Athuraliya, S. Low, V. Li, and Q. Yin, "REM: Active queue management," *IEEE Network*, vol. 15, no. 3, pp. 48–53, 2001.
- [14] T. J. Ott, T. V. Lakshman, and L. H. Wong, "SRED: Stabilized RED," in *Proc. IEEE INFOCOM'99*, vol. 3, New York, NY, Mar. 1999, pp. 1346–1355.
- [15] J. Awaya, M. Ouellette, and D. Y. Montuno, "A control theoretic approach to active queue management," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 36, no. 2-3, pp. 203–235, 2001.
- [16] C. V. Hollot, V. Misra, D. F. Towsley, and W. Gong, "On designing improved controllers for AQM routers supporting TCP flows," in *Proc. IEEE INFOCOM'01*, 2001, pp. 1726–1734.
- [17] K. Claffy, G. Miller, and K. Thompson, "The nature of the beast: Recent traffic measurements from an Internet backbone," in *Proceedings of INET'98*, Geneva, Switzerland, July 1998.
- [18] D. P. F. M. C. Oliveira and J. C. Geromel, *LMI Solver User's Guide*. [Online]. Available: <http://www.dt.fee.unicamp.br/mauricio/lmisol/userguide.ps.gz>